AMENDMENTS TO THE CLAIMS:

This listing of claims replaces all prior versions and listings of claims in the

application:

LISTING OF CLAIMS:

1. (Currently Amended) A ~~network~~ processor comprising:

a command ~~FIFO~~ first-in-first-out (FIFO) configured to store ~~one or more~~ commands, the

commands comprising a first command associated with a first context;

~~a plurality of~~ cores comprising a first core and a second core;

~~a like plurality of same-context command~~ same-context-command FIFOs configured to

store commands of a same context, the same-context-command FIFOs comprising a first same-

context-command FIFO coupled to the first core and a second same-context-command FIFO

coupled to the second core ~~each of said plurality of same context command FIFOs coupled to a~~

~~corresponding one of said first plurality of cores~~; and

a scheduler configured to receive the first command from the ~~coupled to said~~ command

FIFO and to:

if a first core is idle, determine whether a second core is idle;

if the second core is not idle and the second core is processing a second command

associated with a second context, determine whether the second context is the same as the

first context; and

Applicants : Jaroslaw J. Sydir et al.   Attorney's Docket No.: INTEL-012PUS
Serial No. : 10/747,852   Intel Docket No. P17939
Filed   : December 29, 2003
Page   : 3 of 19

if the second context is the same as the first context, store the first command in
the second same-context-command FIFO ~~each of said first plurality of cores in said core~~
~~group~~.

2. (Currently Amended) The ~~network~~ processor of claim 1 wherein the scheduler is
further configured to, if the second core is idle, assign the first command to the first core for
processing

~~said plurality of cores corresponds to a first plurality of cores in a first core group and the~~
~~network processor further comprises a plurality of core groups, each of said core groups coupled~~
~~to the command FIFO~~.

3. (Currently Amended) The ~~network~~ processor of claim 1 wherein ~~said plurality of~~ the
cores correspond ~~corresponds~~ to a first plurality of cores in a first core group and the ~~network~~
processor further comprises:

a plurality of core groups; and

~~said~~ the command FIFO corresponds to a first one of a plurality of command FIFOs, each
of ~~said~~ the plurality of command FIFOs coupled to a corresponding one of ~~said~~ the plurality of
core groups.

Claim 4 (Cancelled)

5. (Currently Amended) The ~~network~~ processor of claim 1 wherein ~~said~~ the scheduler is

further configured to determine if the first core is idle ~~adapted to receive a command from said~~

~~command FIFO and to assign the command to one of said plurality of cores~~.

6. (Currently Amended) The ~~network~~ processor of claim 1 wherein ~~said~~ the scheduler is

further configured to determine whether the second same-context-command FIFO is full ~~adapted~~

~~to receive a command from said command FIFO and to determine which of said plurality of~~

~~cores is processing a command and in response to one of said plurality of cores in said core~~

~~group processing a command, to determine if the command received from said command FIFO~~

~~is in the same context as the command being processed by said core~~.

7. (Currently Amended) A network comprising:

a first node having a ~~network~~ processor, said ~~network~~ processor comprising:

a command ~~FIFO~~ first-in-first-out (FIFO) configured to store ~~one or more~~

commands, the commands comprising a first command associated with a first context;

~~a plurality of~~ cores comprising a first core and a second core;

~~a like plurality of same context command~~ same-context-command FIFOs

configured to store commands of a same context, the same-context-command FIFOs

comprising a first same-context-command FIFO coupled to the first core and a second

same-context-command FIFO coupled to the second core ~~each of said plurality of same~~

~~context command FIFOs coupled to a corresponding one of said first plurality of cores~~;

and

Applicants : Jaroslaw J. Sydir et al.
Serial No. : 10/747,852
Filed : December 29, 2003
Page : 5 of 19

Attorney's Docket No.: INTEL-012PUS
Intel Docket No. P17939

a scheduler configured to receive the first command from the coupled to said

command FIFO and to:

if a first core is idle, determine whether a second core is idle;

if the second core is not idle and the second core is processing a second

command associated with a second context, determine whether the second context

is the same as the first context; and

if the second context is the same as the first context, store the first

command in the second same-context-command FIFO each of said first plurality

of cores in said core group; and

an interface, adapted configured to couple the first node to another node.

8. (Currently Amended) The network of claim 7 further comprising:

a network communication path coupled to the said first node;

a second node having a network processor, the said network processor comprising:

a command FIFO first-in-first-out (FIFO) configured to store one or more

commands, the commands comprising a first command associated with a first context;

a plurality of cores comprising a first core and a second core;

a like plurality of same context command same-context-command FIFOs

configured to store commands of a same context, the same-context command FIFOs

comprising a first same-context-command FIFO coupled to the first core and a second

same-context-command FIFO coupled to the second core each of said plurality of same

Applicants : Jaroslaw J. Sydir et al.   Attorney's Docket No.: INTEL-012PUS
Serial No. : 10/747,852   Intel Docket No. P17939
Filed   : December 29, 2003
Page   : 6 of 19

~~context-command FIFOs coupled to a corresponding one of said first plurality of cores~~;

and

a scheduler <u>configured to receive the first command from the</u> ~~coupled to said~~

command FIFO and to<u>:</u>

<u>if a first core is idle, determine whether a second core is idle;</u>

<u>if the second core is not idle and the second core is processing a second</u>

<u>command associated with a second context, determine whether the second context</u>

<u>is the same as the first context; and</u>

<u>if the second context is the same as the first context, store the first</u>

<u>command in the second same-context-command FIFO</u> ~~each of said first plurality~~

~~of cores in said core group~~; and

an interface coupled to <u>the</u> ~~said~~ first node through <u>the</u> ~~said network~~

communication path.

9. (Original) A processor comprising:

(a) a plurality of cryptographic acceleration units, each of said plurality of cryptographic

acceleration units comprising:

(1) a command first-in-first-out (FIFO) queue to store one or more commands;

(2) a plurality of cores;

(3) a like plurality of same-context command FIFO queues, each of said plurality

of same context command FIFO queues coupled to a corresponding one of said first

plurality of cores; and

Applicants : Jaroslaw J. Sydir et al.   Attorney's Docket No.: INTEL-012PUS
Serial No. : 10/747,852   Intel Docket No. P17939
Filed   : December 29, 2003
Page   : 7 of 19

(4) a scheduler coupled to said command FIFO queue and to each of said first plurality of cores in said core group.

10. (Original) The processor of claim 9 further comprising:

a global FIFO queue having an input adapted to receive commands directed toward at least one of said plurality of cryptographic acceleration units and having an output; and

a global scheduler having an input coupled to the output of said global FIFO queue and having an output adapted to provide a data path to each of said plurality of cryptographic acceleration units.

11. (Original) The processor of claim 10 wherein said plurality of cores form a first core group and wherein the processor further comprises a plurality of core groups, each of plurality of core groups coupled to said global scheduler.

12. (Original) A processor comprising:

a cryptographic acceleration unit comprising:

       (1) a command first-in-first-out (FIFO) queue to store one or more commands;

       (2) a core group comprising a plurality of cores;

       (3) a like plurality of same-context command FIFO queues, each of said plurality of same context command FIFO queues coupled to a corresponding one of said plurality of cores; and

Applicants : Jaroslaw J. Sydir et al.   Attorney's Docket No.: INTEL-012PUS
Serial No. : 10/747,852   Intel Docket No. P17939
Filed : December 29, 2003
Page : 8 of 19

(4) a scheduler coupled to said command FIFO queue and to each of said first

plurality of cores in said core group.

13. (Original) The processor of claim 12 wherein:

said core group corresponds to a first one of a plurality of core groups in said

cryptographic acceleration unit, with each of said plurality of core groups comprising one or

more cores; and said scheduler is coupled to each of said cores.

14. (Original) The processor of claim 13 wherein said cryptographic acceleration unit is

a first one of a plurality of cryptographic acceleration units with each of said plurality of

cryptographic acceleration unit comprising a plurality of core groups with each of said plurality

of core groups comprising one or more cores and a scheduler coupled to each of said cores and

the processor further comprises:

a cross unit scheduler adapted to receive commands and to direct the commands to at

least one of said plurality of cryptographic acceleration units.

15. (Currently Amended) A method comprising:

receiving a <u>first</u> command <u>from</u> in a command F̶I̶F̶O̶ <u>first-in-first-out (FIFO), the first</u>

<u>command associated with a first context</u>;

determining whether <u>a first core</u> o̶n̶e̶-̶o̶f̶-̶a̶-̶p̶l̶u̶r̶a̶l̶i̶t̶y̶-̶o̶f̶-̶c̶o̶r̶e̶s̶ is idle<u>, the first core coupled</u>

<u>to a first same-context-command FIFO configured to store commands of a same context</u>;

   <u>if a first core is idle, determine whether a second core is idle;</u>

if the second core is not idle and the second core is processing a second command

associated with a second context, determine whether the second context is the same as the first

context; and

if the second context is the same as the first context, store the first command in a second

same-context-command FIFO coupled to the second core

~~assigning the command to the idle one of the plurality of cores~~.

16. (Currently Amended) The method of claim 15 further comprising:

if the second core is idle, assign the first command to the first core for processing

~~in response to all of the cores in the core group being idle, assigning the command in the~~

~~command FIFO to a first pre-selected one of the plurality of cores;~~

~~in response to not all of the plurality of cores being idle, determining whether the context~~

~~of the command in the command FIFO is the same as the context of the command currently~~

~~being processed by one of the plurality of cores which is not idle.~~

17. (Currently Amended) The method of claim 16 further comprising:

determining whether the second same-context-command FIFO is full

~~in response to the contexts being the same, moving the command from the command~~

~~FIFO to a same-context-command-buffer associated with the core which is currently processing~~

~~the command in the same context.~~

Applicants : Jaroslaw J. Sydir et al.
Serial No. : 10/747,852
Filed : December 29, 2003
Page : 10 of 19

Attorney's Docket No.: INTEL-012PUS
Intel Docket No. P17939

18. (Currently Amended) The method of claim [[16]] 15 wherein the first core and the

second core are one of a cypher core or an authentication core

~~further comprising:~~

~~in response to the context of the command in the command buffer not being the same as~~

~~the context of the command being processed by the core, assigning the command in the~~

~~command FIFO to a first one of the plurality of cores which is idle.~~

19. (Currently Amended) A method comprising:

~~receiving an indication that at least one core in a core group is idle;~~

~~determining if there are any commands in a command buffer;~~

receiving a command in a command first-in-first-out (FIFO);

determining whether a first core is idle, the first core coupled to a first same-context-

command FIFO configured to store commands of a same context;

if a first core is idle, determining whether a second core is idle;

if the second core is not idle and the second core is processing a second command

associated with a second context, determining whether the second context is the same as the first

context;

if the second context is the same as the first context, determining whether the second

same-context-command FIFO is full;

if the second same-context-command FIFO is not full, store the first command in a

second same-context command FIFO coupled to the second core; and

if the second core is idle, assign the first command to the first core for processing

~~in response to finding a command in the command FIFO, assigning the command to an~~ ~~idle one of the plurality of cores in the core group~~.

20. (Currently Amended) The method of claim 19[[,]] wherein <u>the first core and the second core are cypher cores</u> ~~prior to assigning the command to an idle one of the plurality of cores in the core group, the method comprises determining whether any other core in the core group is not idle~~.

21. (Currently Amended) The method of claim <u>19 wherein the first core and the second core are authentication cores</u> ~~20 further comprising in response to all of the cores in the core group being idle, assigning the command in the command FIFO to a first pre-determined one of the plurality of cores in the core group~~.

Claims 22 to 24 (Cancelled)

25. (Currently Amended) An article comprising:

a storage medium having stored thereon instructions that when executed by a machine result in the following:

~~scheduling a packet command for processing by one of a plurality of cores~~

~~depending upon whether a core is idle and whether the packet command has the same~~

~~context of a command currently being processed by another one of the plurality of cores;~~

~~and~~

Applicants : Jaroslaw J. Sydir et al.   Attorney's Docket No.: INTEL-012PUS
Serial No. : 10/747,852   Intel Docket No. P17939
Filed   : December 29, 2003
Page   : 12 of 19

~~if the packet command has the same context of a command currently being processed by another one of the plurality of cores, storing the command in a same-context-command FIFO associated with the one of the plurality of cores~~

receiving a first command in a command first-in-first-out (FIFO), the first command being associated with a first context;

determining whether a first core is idle, the first core coupled to a first same-context-command FIFO configured to store commands of a same context;

if a first core is idle, determine whether a second core is idle;

if the second core is not idle and the second core is processing a second command associated with a second context, determine whether the second context is the same as the first context; and

if the second context is the same as the first context, store the first command in a second same-context-command FIFO coupled to the second core.

26. (Currently Amended) The article of claim 25 wherein ~~the storage medium has stored thereon instructions that when executed by a machine result moving the command from the same-context-command FIFO to the core for processing~~ the first core and the second core are one of a cypher core or an authentication core.

27. (New) The processor of claim 1 wherein the first core and the second core are one of a cypher core or an authentication core.

Applicants : Jaroslaw J. Sydir et al.
Serial No. : 10/747,852
Filed : December 29, 2003
Page : 13 of 19

Attorney's Docket No.: INTEL-012PUS
Intel Docket No. P17939

28. (New) The network of claim 7 wherein the first core and the second core are one of a cypher core or an authentication core.

29. (New) The processor of claim 9 wherein the plurality of cores is one of a plurality of cypher cores or a plurality of authentication cores.

30. (New) The processor of claim 12 wherein the plurality of cores is one of a plurality of cypher cores or a plurality of authentication cores.